Clementine 探査機のデータ：その概要と利用法

中野 司*・田中 明子*

Data Acquired by the Clementine Spacecraft

Tsukasa NAKANO* and Akiko TANAKA*

キーワード：Clementine 探査機，月，重力・地形データ，球面調和関数，マルチスペクトル画像，CD-ROM，モザイク画像，ステレオ視画像。

Key Words: Clementine spacecraft, Moon, Gravity and topography data, Spherical Harmonics, Multispectral images, CD-ROM, Mosaic images, Stereoscopic images.

1. はじめに

1994年1月に Vandenburg 空軍基地（アメリカ合衆国）から打ち上げられた Clementine 探査機は，同年2月から5月の71日間に月（第1図）を極軌道で周回し詳細な観測を行った（Nozette *et al.*, 1994）．姿勢制御用コンピュータの故障のために，当初予定されていた小惑星（asteroid）Geographos とのフライバイは行われなかったが，Clementine 探査機は，Apollo 計画以降は惑星・衛星探査機による十分な観測のなかった月についての多数のデータの取得に成功した．

Clementine 探査機が取得したデータは，月のグローバルな重力・地形データと，ほぼ全域の月面を撮影した多数の高精度マルチスペクトル・デジタル画像データである．現時点では，これらのデータの一部がインターネットを通して，もしくはデモ版の CD-ROM として入手可能である．本文では，われわれが入手することのできた Clementine 探査機のデータについて，その入手法やデータの詳細，そしてそれらに対して行った簡単な解析処理の原理や手法などを紹介する．

2. 月の重力データと地形データ

Clementine ミッションでは，地球の追跡ステーションからの観測および天体力学的計算による詳細な探査機の軌道解析から，月の重力ポテンシャル場のデータが求められた．また，これに加えて，探査機に積み込まれた LIDAR（Light Imaging Detection And Ranging）と呼ばれるレー

ザー測距計のデータを用いて，月面の地形高度の分布が得られた（Zuber *et al.*, 1994）．

これらのデータをコンパイルした結果は，有限次数の球面調和関数展開（後述）で近似された以下のような月の重力・地形モデルとして公表されている（Zuber *et al.*, 1994）：

GLGM-1（Goddard Lunar Gravity Model-1）

GLTM-1（Goddard Lunar Topography Model-1）

これらのモデル（正確にはその改良版）の球面調和関数展開の係数データは，下記のインターネットサイトから anonymous FTP で取得可能である．

ftp@utopia.eps.jhu.edu:pub

また，Clementine 探査機が LIDAR を用いてサンプリングした地形高度分布の生データ（サンプリング点での地形高度のデータ；後述）も，このサイトから取得することができる．第2図は，これらのデータから描いた月面の重力異常と地形を示している．以下では，このサイトから取得した重力・地形データについて説明する．

2.1 重力データ

Clementine 探査機によって得られたデータを，それ以前に得られた Lunar Orbiter 1~5 や Apollo-15 subsatellite のデータとあわせてコンパイルした月の重力場モデルの最新版は，GLGM-2 と呼ばれているものらしい（1995年6月末現在；Lemoine *et al.*, 1995）．それは，先に示したインターネットサイトにおかれている GLGM2.FREEAIR というファイルに，70次までの球面調和関数で展開・近似した月の重力ポテンシャルの乱れ $\Delta U(r, \lambda, \phi)$ の係数 C_{nm} , S_{nm} のデータとしておさめられている：

* 地質調査所 Geological Survey of Japan,
1-1-3, Higashi, Tsukuba 305, Japan

第2図. Clementine 探査機の得た月の重力・地形データ。(a) 月のブーゲ重力異常。球面調和関数で表現された月の重力ポテンシャルモデル (GLGM-2, Lemoine *et al.*, 1995; ファイル GLGM2.FREEAIR, JHU, 1995) および地形モデル (GLTM-2, Smith *et al.*, 1995; ファイル GLTM2.TOPO, JHU, 1995) から計算した。(b) GLTM2.TOPO のデータから計算した月の地形モデル。月面画像のモザイク画像 (ファイル ALBEDO.CYLINDER, USGS, 1994) を用いて陰影づけを行った。(c) 月の地形。Clementine 探査機が得た月面地形高度のサンプリングデータ (ファイル GLTM2.DAT; 第3図参照) からスプライン補間 (Briggs, 1974) によって計算した。(b)と同じ方法で陰影づけをしてある。

第3図. Clementine 探査機の軌道解析と LIDAR (レーザー測距計) による計測で得られた月面地形高度のサンプリングデータ (ファイル GLTM2.DAT, JHU, 1995) の空間分布. データ点の総数は 72078.

$$\begin{pmatrix} C'_{nm} \\ S'_{nm} \end{pmatrix} = \begin{pmatrix} C_{nm} \\ S_{nm} \end{pmatrix} - \frac{3}{2n+1} \frac{\rho_c}{\rho_o} \frac{1}{R} \begin{pmatrix} A_{nm} \\ B_{nm} \end{pmatrix} \quad (6)$$

を使って計算できる (たとえば, Phillips and Lambeck, 1980). ここで, ρ_c と ρ_o はそれぞれ月の地殻と月全体の平均密度である (ただし, 第2(a)図では $\rho_c = 2800 \text{ kg/m}^3$ と仮定している). 同様に, 地殻の厚さも表面密度層近似から求めることができる (Zuber *et al.*, 1994).

前述のインターネットサイトには, Clementine 探査機の軌道解析と LIDAR による測距で得られた月の地形高度のサンプリングデータ (点データ) の入ったファイル GLTM2.DAT もおかれていた. このファイルには, 月面上のおおよそ 7 万 2 千点における以下のようなデータがおさめられている:

サンプリング時刻

経度と緯度

GLTM-2 と同じ回転楕円体を基準とした月の地形高度

半径 1738 km の球を基準とした月の地形高度

これらのサンプリング点の月面上での分布を示したものが第3図である. この図より, 高緯度地域にはサンプリングデータが存在しないことがわかる.

ファイル GLTM2.DAT のデータを用いて, GLTM-2 と同じ回転楕円体を基準とした月の地形を, スプライン補間 (Briggs, 1974) によって求めた (第2(c)図). この図では, 高緯度地域ではサンプリング点が存在しない (第3図) ために異常な高度となっているが, それ以外の地域では,

GLTM-2 よりもやや解像度のよい地形分布が示されている.

3. 月面のマルチスペクトル画像データ

Clementine 探査機は, それに積み込まれた 6 台のカメラとそれぞれのカメラに備えられた多数のフィルターホイール (第1表) を用いて, 200 万枚にもものぼるデジタル・マルチスペクトル画像を撮影した (Nozette *et al.*, 1994). これまでに Apollo ミッションなどで撮影されたもの (たとえば, Heiken *et al.*, 1991) と比較すると, 両極や月の裏側 (far side) を含む広い地域の高分解能画像が得られたようである.

1995 年 5 月現在で, これらの画像の一部は以下のインターネットサイトから取得可能である.

<ftp://clementine.sl.gov:pub/clementine>

<http://www.nrl.navy.mil/clementine/clib>

<http://www.flagsvr.wr.usgs.gov:80/clementine>

ただし, 最初のサイトは anonymous FTP でログイン可能な, そして後 2 者は WWW (World Wide Web) のサイトである. また, これらに加えて, Voyager や Magellan 探査機などの場合 (中野, 1994, 1995) と同様にして, NASA から CD-ROM としてすべての Clementine 画像が出版される予定である (全 88 巻; USGS のアナウンスメント, 1995 年 4 月). 以下では, それに先だってデモ版として NASA の共同研究者に配布された CD-ROM について説明する.

第 1 表. Clementine 探査機の撮像系（カメラ・フィルタの種類および画像のサイズ）. Clementine CD-ROM のファイル /index/imgindx.lbl など（第 2 表）に記されていた説明に，Clementine CD-ROM 38 巻におさめられていた画像枚数などをつけ加えた．

カ メ ラ		フ ィ ル タ			画 像	
略 称	正 式 名 称	略 称	中 央 波 長 (nm)	波 長 帯 幅 (nm)	画 像 サイズ (pixels)	CD-ROM 38 巻 中 の 画像枚数†
A-STAR	A STAR tracker camera	(A)	broad band		384×576	0
B-STAR	B STAR tracker camera					0
HIRES	LIDAR High RESolution camera	A	415	40	384×288	0
		B	560	50		0
		C	650	50		0
		D	750	50		9232
		E	650	350		0
		F	opaque (no filter)			0
UVVIS	UltraViolet/VISible camera	A	415	40	384×288	1666
		B	750	10		1666
		C	900	20		1664
		D	950	30		1663
		E	1000	30		1663
		F	650	550		0
NIR	Near InfraRed camera	A	1100	60	256×256	944 (920)
		B	1250	60		944 (921)
		C	1500	60		943 (922)
		D	2000	60		945 (921)
		E	2600	60		948 (921)
		F	2780	60		944 (920)
LWIR	Long Wavelength InfraRed camera	(A)	8750	1500	128×128	4686 (4619)

† かつこ内に示したものはターゲットとして月面を撮影した画像の枚数．

3.1 Clementine CD-ROM

われわれの入手した Clementine 探査機による画像データの入ったデモ版 CD-ROM の正式名称は以下の通りである：

DoD and NASA
Mission to the moon,
Deep space program science experiment
Clementine, EDR image archieve
VOLUME_ID=CL_0038, DEMO version,
Revolutions:170 to 174

これは，今後出版される全88巻のシリーズのうち38巻に相当するもので，そのディレクトリ構成などは今後出版されるものとほとんど変わらないと思われる．

ところで，Clementine CD-ROM のファイル名にはそのバージョン番号が”；”とともにつけ加わっているため，一部の UNIX ワークステーションではコマンドラインからの CD-ROM 上のファイル名の指定がめんどうである（UNIX では”；”をコマンドラインのマルチステートメントの区切りとして用いるため，それをファイル名の一部として持つ場合には” ’ ”などでファイル名全体を囲んで指定をする必要がある）．また，一般に CD-ROM 上のファイル・ディレクトリ名が小文字になるか大文字になるかは各社の UNIX によってまちまちであるため，様々なワークステーションで稼働する CD-ROM 用のソフトウェアの

作成が困難である．そこで，われわれはこれらの難点を回避するために，CD-ROM 上のすべてのファイル・ディレクトリの名前だけを，バージョン番号を取り去った小文字の名前としてハードディスクにコピーする（正確にはシンボリックリンクする）ための UNIX 用プログラム link.cdrom を利用している．このプログラムを用いれば，CD-ROM 上のファイルの内容をコピーすることなしに，それらがあたかも通常のハードディスク上の小文字名のファイルであるかのようにアクセスできる．付録 C にソースリストを示したプログラム link.cdrom および，それを高速化した linkcdrom の使い方は以下の通りである．

link.cdrom CD-ROM_directory HD_directory
もしくは

linkcdrom CD-ROM_directory HD_directory

ここで，CD-ROM_directory は CD-ROM がマウントされているディレクトリ名（たとえば”/cdrom”；システム・環境に依存），HD_directory はファイル・ディレクトリの名前をコピー（リンク）したいハードディスク上の適当なディレクトリ名である．なお，これによって作成したディレクトリ・ファイルに対しても，通常のものと同様なコピー・消去などが行える（当然のことながら，リンクしたファイル名を消去しても CD-ROM 上の内容まで消されるわけではない）．以下の説明では，このようにしてリンクしたディレクトリ・ファイル名を用いることにする．

第2表にデモ版の Clementine CD-ROM の概要を示す。この表からもわかるように Clementine CD-ROM では、探査機の月周回軌道 (revolution) ごとにディレクトリに分類されておさまられている画像データと、多数のドキュメントおよびソフトウェア類が入っている。画像データの利用法については後述することにして、ここでは CD-ROM 上のドキュメントとソフトウェアについてコメントする。

Clementine CD-ROM におさまられたドキュメント類 (第2表) のうちディレクトリ /document のものは、単純なテキストファイル以外に Microsoft Word で読み込める形式のものと EPS (Encapsulated Postscript) で記述されたものがあった (それらの説明はファイル docinfo.txt に記されている)。これらのドキュメント類は懇切丁寧で、内容も Clementine ミッションの概略、探査機のシステム、画像データベースや画像フォーマット・ソフトウェアの説明など多岐に渡っている (本文もこれらに負うところが大きい)。

ディレクトリ /software にあるソフトウェアは、第2表のディレクトリ名に相当する3つの OS で起動可能な画像ファイルの解凍と画像表示のためのものである。前者は JPEG 圧縮 (後述) された画像を NASA がこれまでの CD-ROM で用いてきた PDS フォーマット (中野, 1995) などに変換するものであり、後者は、NIH Image (Apple Macintosh 用) や xv (UNIX 用) などのフリーソフトウェアである。それぞれのディレクトリには、C 言語で書かれたソースコードや実行可能なオブジェクトコード、そして

それらの説明書などが入っていた。なお、ついでながら、ディレクトリ /sunos におさまられている Clementine 画像の解凍プログラム clemdcmp のソースコードは、ANSI-C を処理可能であれば SunOS 以外でもコンパイルして使用可能である (ただし、コンパイル時に "-Dsun" という指定を行う必要がある)。

3.2 画像ファイルとインデックスファイル

Clementine CD-ROM の画像データは、月周回軌道ごとディレクトリに分類されて、画像を得たカメラやフィルタの種類によって命名されたファイルにおさまられている (画像の命名法は /document/arcsis.* で詳しく説明されている)。これらのファイルは、PDS (Planetary Data System) ラベルと呼ばれる画像の説明・記述部と JPEG 圧縮された画像データ部からなる (これらの説明は /document/edrsis.* に詳しい)。JPEG 圧縮された画像データは、Clementine 探査機に登載された CNES (French Space Agency) の開発したシステムの出力そのものではないかと考えられる。このような画像の圧縮率はかなりよく (およそ 27%; デモ版の CD-ROM の全画像の平均値)、そのために1枚の CD-ROM に、およそ2万8千枚 (デモ版の CD-ROM の場合) にもよる画像データファイルがおさまられている (第1, 2表)。

画像データファイルの PDS ラベルには、これまで NASA から出版されている惑星・衛星の光学画像データに添付されていたもの (中野, 1995) よりもずっと高精度・詳細な撮影時の情報など (後述) が記述されている。また、すべての画像の PDS ラベルの情報はまとめられて、/index/imgindx.tab という名前で画像インデックスファイルとして CD-ROM におさまられている。それには、CD-ROM に含まれるすべての画像について、第3表に示したような項目のデータが各々882バイトのレコードとしておさまられている (第3表に示したそれぞれの項目の意味は、/index/imgindx.1b1 というファイルにおいて詳しく説明されている)。

3.3 画像データの利用法

前述した画像インデックスファイルを用いれば、Clementine CD-ROM 中の画像の検索・表示を行うための画像データベースを容易に構築できる。さらに、それに含まれる諸情報を利用すれば、画像中の各画素の月面上での位置を決めること (位置決め)、そして、その手法を応用して多数の画像を張り合わせたモザイク画像を作成することなどが可能である。さらに、画像撮影時の探査機 (カメラ) の位置を考慮してステレオ視が可能な画像ペアを検索することもできる。Clementine の画像データの位置決めの手法、

第2表. Clementine CD-ROM の概要. デモ版 (38巻) の CD-ROM の内容.

voldesc.cat	CD-ROM のコンテンツシート.
aareadme.txt	Clementine CD-ROM の概略説明書.
errata.txt	Clementine CD-ROM セット全体のエラー情報.
/document	ドキュメント類 (13 files) のためのディレクトリ.
docinfo.txt	ドキュメント類の説明書.
volinfo.*	ミッションの概略と探査機の撮像系の説明書.
arcsis.*	画像データベース (CD-ROM セット) の説明書.
edrsis.*	画像ファイルの説明書.
/software	画像を取り扱うためのソフトウェア類 (12 directories, 28 files) の入ったディレクトリ.
/pcdos	PCDOS 用のソフトウェア類.
/mac	MacOS 用のソフトウェア類.
/sunos	UNIX (SunOS) 用のソフトウェア類.
/timeline	Clementine の航行計時に関する情報 (6 files) の入ったディレクトリ.
/index	CD-ROM 中の画像のインデックスファイル類 (7 files; total ~24M bytes) の入ったディレクトリ.
imgindx.*	CD-ROM の全画像のインデックスファイル.
missindx.*	CD-ROM 中のエラー画像のインデックス.
/lunXXX	番号 XXX の周回軌道において撮影した画像の入ったディレクトリ. 以下に示すものは CD-ROM 38巻 (デモ版) の内容.
/lun170	77 directories, 5532 files; total ~112M bytes.
/lun171	65 directories, 5320 files; total ~104M bytes.
/lun172	77 directories, 5636 files; total ~112M bytes.
/lun173	79 directories, 5789 files; total ~114M bytes.
/lun174	77 directories, 5631 files; total ~111M bytes.

第3表. Clementine CD-ROM 上の画像インデックスファイル /index/imgindx.tab (第2表) におさめられているデータ項目の名称, およびそれらの項目各々の各レコード (CD-ROM 上の各画像に対応; 1レコードは 882 バイト) 上での先頭の位置とデータ長 (単位はバイト). これらのデータ項目の詳細な説明は CD-ROM の /index/imgindx.lbl などに記されている.

	項 目 名	先頭	長さ		項 目 名	先頭	長さ
1	VOLUME_ID	2	7	38	RETICLE_POINT_LATITUDE_1	437	7
2	REVOLUTION_NUMBER	11	3		RETICLE_POINT_LATITUDE_2	453	7
3	FRAME_SEQUENCE_NUMBER	15	5		RETICLE_POINT_LATITUDE_3	469	7
4	ORIGINAL_FRAME_SEQUENCE_NUMBER	21	5		RETICLE_POINT_LATITUDE_4	485	7
5	INSTRUMENT_ID	28	6	39	RETICLE_POINT_LONGITUDE_1	445	7
6	LINES	36	3		RETICLE_POINT_LONGITUDE_2	461	7
7	SAMPLES	40	3		RETICLE_POINT_LONGITUDE_3	477	7
8	UNCORRECTED_START_TIME	45	24		RETICLE_POINT_LONGITUDE_4	493	7
9	START_TIME	72	24	40	HORIZONTAL_PIXEL_SCALE	501	7
10	EXPOSURE_DURATION	98	10	41	VERTICAL_PIXEL_SCALE	509	7
11	GAIN_MODE_ID	110	3	42	SMEAR_MAGNITUDE	517	7
12	MCP_GAIN_MODE_ID	116	3	43	SMEAR_AZIMUTH	525	7
13	OFFSET_MODE_ID	122	3	44	NORTH_AZIMUTH	533	7
14	FILTER_NAME	128	1	45	SUB_SPACECRAFT_LATITUDE	541	7
15	MEAN	131	10	46	SUB_SPACECRAFT_LONGITUDE	549	7
16	STANDARD_DEVIATION	142	10	47	SPACECRAFT_ALTITUDE	557	8
17	MAXIMUM	153	3	48	SUB_SPACECRAFT_AZIMUTH	566	7
18	MINIMUM	157	3	49	SC_SUN_POSITION_VECTOR_1	574	12
19	ENCODING_COMPRESSION_RATIO	161	5		SC_SUN_POSITION_VECTOR_2	587	12
20	ORIGINAL_PRODUCT_ID	168	12		SC_SUN_POSITION_VECTOR_3	600	12
21	FILE_NAME	183	12	50	SC_SUN_VELOCITY_VECTOR_1	613	8
22	PRODUCT_ID	183	12		SC_SUN_VELOCITY_VECTOR_2	622	8
23	DOWNLOAD_ID	198	8		SC_SUN_VELOCITY_VECTOR_3	631	8
24	SEQUENCE_TABLE_ID	209	16	51	SPACECRAFT_SOLAR_DISTANCE	640	11
25	CENTRAL_BODY_NAME	228	5	52	SOLAR_DISTANCE	652	11
26	TARGET_NAME	236	5	53	SUB_SOLAR_AZIMUTH	664	7
27	LIGHT_SOURCE_NAME	244	5	54	SUB_SOLAR_LATITUDE	672	7
28	RIGHT_ASCENSION	251	7	55	SUB_SOLAR_LONGITUDE	680	7
29	DECLINATION	259	7	56	TWIST_ANGLE	688	7
30	RETICLE_POINT_RA_1	267	7	57	INCIDENCE_ANGLE	696	7
	RETICLE_POINT_RA_2	283	7	58	PHASE_ANGLE	704	7
	RETICLE_POINT_RA_3	299	7	59	EMISSION_ANGLE	712	7
	RETICLE_POINT_RA_4	315	7	60	LOCAL_HOUR_ANGLE	720	7
31	RETICLE_POINT_DECLINATION_1	275	7	61	LIGHT_SOURCE_DISTANCE	728	11
	RETICLE_POINT_DECLINATION_2	291	7	62	SUB_LIGHT_SOURCE_AZIMUTH	740	7
	RETICLE_POINT_DECLINATION_3	307	7	63	SUB_LIGHT_SOURCE_LONGITUDE	748	7
	RETICLE_POINT_DECLINATION_4	323	7	64	SUB_LIGHT_SOURCE_LATITUDE	756	7
32	SC_TARGET_POSITION_VECTOR_1	331	12	65	LIGHT_SOLAR_INCIDENCE_ANGLE	764	7
	SC_TARGET_POSITION_VECTOR_2	344	12	66	LIGHT_SOURCE_PHASE_ANGLE	772	7
	SC_TARGET_POSITION_VECTOR_3	357	12	67	PATH_NAME	781	25
33	SC_TARGET_VELOCITY_VECTOR_1	370	8	68	MISSION_PHASE_NAME	809	22
	SC_TARGET_VELOCITY_VECTOR_2	379	8	69	ENCODING_TYPE	834	12
	SC_TARGET_VELOCITY_VECTOR_3	388	8	70	LENS_TEMPERATURE	848	8
34	TARGET_CENTER_DISTANCE	397	11	71	FOCAL_PLANE_TEMPERATURE	857	8
35	SLANT_DISTANCE	409	11	72	CRYOCOOLER_TEMPERATURE	866	8
36	CENTER_LATITUDE	421	7	73	CRYOCOOLER_DURATION	875	6
37	CENTER_LONGITUDE	429	7				

overlap= 0.63295, angle= 3.00071 degree
Slave : lun171/lunxxxxx/lunxxxxx/inf0375e.171 (NIR/F)
Master: lun171/lunxxxxx/lunxxxxx/inf0344e.171 (NIR/F)
corner (Lon., lat.):= (350.99, -48.31), (351.21, -45.20),
(degree) (355.67, -48.37), (355.63, -45.25).
width= 94.389 km, height= 94.2421 km

overlap= 0.605133, angle= 3.19839 degree
Slave : lun171/lunxxxxx/lunxxxxx/inf0468e.171 (NIR/F)
Master: lun171/lunxxxxx/lunxxxxx/inf0437e.171 (NIR/F)
corner (Lon., lat.):= (351.23, -44.97), (351.40, -42.01),
(degree) (355.42, -45.03), (355.40, -42.07).
width= 89.6848 km, height= 89.859 km

第7図. Clementine CD-ROM 38巻 (デモ版) に含まれる画像のうちステレオ視が可能なおペアの例. どちらのおペアも Tycho クレーターの東南に隣接する地域の NIR カメラ・F フィルタ (第1表参照) で得られた画像である (A, B で示したクレータは第6図にも含まれている; ただし, ここに示した画像は第6図と異なり右方向が北である). ステレオペアの上部に記したパラメータのうち corner (画像の四隅) および width・height (画像の横・縦幅) は, 画像の隅に M と記された Master 画像 (付録 B 参照) に対する値である. また, overlap および angle は, 付録 B で説明している2枚の画像の重なり具合とそれらに対する視線ベクトル間の角度 θ を示している.

第6図. Clementine 探査機の近赤外 (NIR) カメラの F フィルタ (第1表) で撮影された Tycho クレータ周辺の画像をつなぎ合わせたモザイク画像. 画像の横・縦幅はそれぞれ 250 および 400 km である.

(同じ項目を重複して指定してもよい)。そして、ファイルの終了、もしくは項目データの位置と長さ指定がともに0なら記述は終わりと見なされる。なお、tab2tbl は、このファイル上の“#”から改行までを読み飛ばすので、そこに適当なコメントを記述することができる。前述したように、このようにして起動した tab2tbl は、画像インデックスファイルの指定した項目データをタブコードで区切り、レコード(画像)ごとに1行にまとめて出力する。この出力は、UNIX(もしくはMS-DOS用)のパターン処理言語 awk で取り扱うと便利である。

tab2tbl のソースコードおよび、それを応用した簡便な Clementine CD-ROM の画像データ検索プログラムの例を付録Cに示した。付録Cに示されているUNIXのC-Shell用スクリプト count.cf は、プログラム起動時のパラメータとして与えられたカメラとフィルタ(第1表の略称を指定)で得られたCD-ROM上の画像データファイルの個数を数え上げる(正確には、それらの画像データのうちターゲットとして月面の像を含むものとそうでないものを区別して数え上げる)。また、スクリプト search.loc は、パラメータとして指定された経度・緯度(度単位)の月面上の点(第1, 2図)を含む Clementine 画像を検索するために利用することができる。

4. おわりに

Clementine 探査機の得たデータは、Voyager や Viking Orbiter などのこれまでの惑星・衛星探査機の得たデータ(中野, 1995)に比べて非常に高精度である。特に画像データは、月面のほとんどの地域について高分解能であることは言うに及ばず、高精度の位置決めが可能なマルチスペクトル画像であるため、異なったバンド(波長帯)の画像を組み合わせ、J-ERS-1 や LANDSAT のような地球のリモートセンシング衛星の画像データと同様な手法で、ローカルもしくはグローバルな月面物質の推定におおいに利用できる(たとえば, Pieters *et al.* (1994) や Lucey *et al.* (1995) の解析を参照のこと)。

Clementine ミッションは軍事的な目的があったこと(Clementine 探査機は超軽量テクノロジーの産物であるらしい; Nozette *et al.*, 1994), そして計画が非常に速く進行したこと(1990年に計画が本格的にスタートした; Clementine CD-ROM のファイル /document/volinfo.txt の記述)などの理由で、事前のアナウンスメントがあまりなかった。そのためか、同時期の Magellan 探査機のデータの場合(中野, 1994)と比較すると、現時点では取得されたデータを用いた研究成果はそう多くは発表されていないようである。今後出版される CD-ROM の画像データに対して本文で紹介したような手法を用いて解析を行えば、

オリジナルな研究の余地は十分残っているものと思われる。このような惑星地質学の研究、そして来年以降にわが国が行う LUNA-A (月探査ミッション) や PLANET-B (火星探査ミッション) 計画の事前勉強に本文が一助となれば幸いである。

謝 辞

Clementine の諸データの入手法についての情報は、寺藺淳也氏(宇宙科学研究所, 現宇宙開発事業団)と山口靖氏(地質調査所)に御教示いただいた。Clementine CD-ROM (38巻; デモ版)は高田淑子氏(東京大学理学部)からお借りした。Clementine ミッションの成功とそれによって得られた諸データのインターネットや CD-ROM による一般への配布は、アメリカ合衆国の以下の機関の努力の結果である。

DoD (Department of Defense)

BMDO (Ballistic Missile Defense Organization)

NRL (Naval Research Laboratory)

LLNL (Lawrence Livermore National Laboratory)

NASA

(National Aeronautics and Space Administration)

JPL (Jet Propulsion Laboratory)

GSFC (Goddard Space Flight Center)

USGS (United States Geological Survey)

JHU (John Hopkins University)

ここに記してこれらの方々に感謝いたします。

後 記

本文を執筆後に、アメリカ合衆国 NSSDC (National Space Science Data Center) から Clementine CD-ROM の出版についての正式なアナウンスメントがあった。本文で紹介したものと同様な CD-ROM (全88巻)の一般への配布が1995年7月から始まったとのことである。NSSDC からそれら入手する方法については、白尾(1995)の説明を参照していただきたい。

文 献

Clementine 画像がおさめられた CD-ROM とそれに含まれるドキュメント類については本文中に記したので、ここでは割愛する。

Briggs, I. C. (1974) Machine contouring using minimum curvature, *Geophys.*, 39, 39-48.

Heiken, G. H., Vaniman, D. T. and French, B. M. ed. (1991) *Lunar sourcebook*, Cambridge Univ. Press, 736pp.

Lemoine, F. G., Smith, D. E., Zuber, M. T., Neumann, G.

- A. and Rowlands, D. D. (1995) A 70th degree and order lunar gravity model from Clementine and historical data, submitted to *J. Geophys. Res.*
- Lucey, P. G., Taylor, G. J. and Malaret, E. (1995) Abundance and distribution of iron on the moon, *Science*, 268, 1150-1153.
- 中野 司 (1994) Magellan CD-ROM のデータ, 地質ニュース, No. 473, 50-57.
- 中野 司 (1995) 惑星・衛星のデジタル画像データの利用法, 惑星火山学入門 (藤井直之・白尾元理・小森長生編, 日本火山学会 月・惑星火山ワーキンググループ, 176 pp), 115-130.
- Nozette, S., Rustan, P., Plesance, L. P., Horan, D. M., Regeon, P., Shoemaker, E. M., Spudis, P. D., Acton, C. H., Baker, D. N., Blamont, J. E., Buratti, B. J., Corson, M. P., Davies, M. E., Duxbury, T. C., Eliason, E. M., Jakosky, B. M., Kordas, J. F., Lewis, I. T., Lichtenberg, C. L., Lucey, P. G., Malaret, E., Massie, M. A., Resnick, J. H., Rollins, C. J., Park, H. S., McEwen, A. S., Priest, R. E., Pieters, C. M., Reisse, R. A., Robinson, M. S., Simpson, R. A., Smith, D. E., Sorenson, T. C., Vorder Breugge, R. W. and Zuber, M. T. (1994) The Clementine mission to the moon: Scientific overview, *Science*, 266, 1835-1839.
- Phillips, R. J. and Lambeck, K. (1980) Gravity fields of the terrestrial planets: Long-wavelength anomalies and tectonics, *Rev. Geophys.*, 18, 27-76.
- Pieters, C. M., Staid, M. I., Fischer, E. M., Tompkins, S. and He, G. (1994) A shaper view of impact craters from Clementine data, *Science*, 266, 1844-1848.
- 力武常次・佐藤良輔・萩原幸男 (1980) 物理数学 I, 学会出版センター, 163pp.
- 白尾元理 (1995) 惑星火山学のデータ入手法, 惑星火山学入門 (藤井直之・白尾元理・小森長生編, 日本火山学会 月・惑星火山ワーキンググループ, 176pp), 167-176.
- Smith, D. E., Zuber, M. T., Neumann, G. A. and Lemoine, F. G. (1995) The topography of the Moon from the Clementine LIDAR, submitted to *J. Geophys. Res.*
- Zuber, M. T., Smith, D. E., Lemoine, F. G. and Neumann, G. A. (1994) The shape and internal structure of the moon from the Clementine mission, *Science*, 266, 1839-1843.

付録 A. Clementine 画像の位置決め

以下では, CD-ROM として配布された Clementine 探査機の得た画像上の点の座標からその点に対応するターゲット

ト (月) 上の点の座標を求める (およびその逆の計算を行う) ための画像の位置決めの計算式について説明する. なお, ここで示されている位置決めの手法は, Voyager や Viking Orbiter などの惑星・衛星探査機で得られた光学画像に対しても広く適用可能である (中野, 1995). 以下の説明中で用いられている Clementine 探査機とターゲットの位置関係やそれらを表すベクトル記号, そして計算に必要なデータの CD-ROM 上での所在などについては第 4 図および第 3 表を参照していただきたい.

A.1 仮定

- (1) ターゲット (月) は半径 R の球であるとする. ただし, R はあらかじめ与えられたものでなく, 与えられたデータから計算で求めるものとする.
- (2) スクリーン (画像) は平らで画素の形状は長方形である (魚眼レンズのようなものを用いて得られた画像でない) とする. また, Clementine 探査機のカメラの軸線はスクリーン中央の点 C' でスクリーンと直交していると考え. なお, 焦点距離やスクリーン上のスケールなどの撮像系に関するパラメータはあらかじめ与えられていない (与えられたデータから計算で求める) ものとする.

A.2 与えられたデータ (第 4 図および第 3 表を参照のこと)

SAMPLES, H:

LINES, V:

Clementine 画像の横・縦の画素数.

TARGET_CENTER_DISTANCE, T:

Clementine 探査機 (カメラ) S とターゲットの中心 O の間の距離.

SLANT_DISTANCE, L:

カメラの軸線 SC とターゲットの表面の交点 C とカメラ S との間の距離.

SUB_SPACECRAFT_LONGITUDE, λ_p :

SUB_SPACECRAFT_LATITUDE, ϕ_p :

カメラとターゲット中心を結ぶ線 OS とターゲットの表面の交点 P の経度と緯度.

CENTER_LONGITUDE, λ_c :

CENTER_LATITUDE, ϕ_c :

カメラの軸線 SC とターゲットの表面の交点 C の経度と緯度.

RETICLE_POINT_LONGITUDE_1, λ_1 :

RETICLE_POINT_LATITUDE_1, ϕ_1 :

画像の左上隅の点 Q'_1 に対応するターゲット上の点 Q_1 の経度と緯度.

RETICLE_POINT_LONGITUDE_2, λ_2 :

RETICLE_POINT_LATITUDE_2, ϕ_2 :

画像の右上隅の点 Q'_2 に対応するターゲット上の点 Q_2 の経度と緯度.

RETICLE_POINT_LONGITUDE_3, λ_3 :

RETICLE_POINT_LATITUDE_3, ϕ_3 :

画像の右下隅の点 Q'_3 に対応するターゲット上の点 Q_3 の経度と緯度.

RETICLE_POINT_LONGITUDE_4, λ_4 :

RETICLE_POINT_LATITUDE_4, ϕ_4 :

画像の左下隅の点 Q'_4 に対応するターゲット上の点 Q_4 の経度と緯度.

A.3 ターゲット（月）の半径の算出

ターゲットの中心 O からカメラ S へ向かうベクトル,

$$\vec{t} = T \begin{pmatrix} \cos \phi_p & \cos \lambda_p \\ \cos \phi_p & \sin \lambda_p \\ \sin \phi_p \end{pmatrix} \quad (A-1)$$

と O からターゲット表面の点 C に向かうベクトル,

$$\vec{r}_c = R \begin{pmatrix} \cos \phi_c & \cos \lambda_c \\ \cos \phi_c & \sin \lambda_c \\ \sin \phi_c \end{pmatrix} \quad (A-2)$$

のなす角を θ とすると, 三角形 SOC に余弦定理を用いてターゲットの半径 R は,

$$\begin{aligned} \cos \theta &= \vec{t} \cdot \vec{r}_c / (T R), \\ R &= T \cos \theta - (L^2 - T^2 \sin^2 \theta)^{1/2} \end{aligned} \quad (A-3)$$

となる.

A.4 スクリーン上の点の位置の表現

カメラ S からスクリーンの中央の点 C' に向かうベクトルは, スクリーン上で C' が S からターゲットの表面の点 C を見通した点であることから,

$$(\alpha/L) \vec{l}_c \quad (A-4)$$

と表すことができる. ただし, α はカメラレンズの焦点距離で, 与えられたデータから算出することができる (後述). また, $\vec{l}_c$ は S から C に向かう長さ L のカメラの軸線ベクトルで, 以下のように表すことができる.

$$\vec{l}_c = \vec{r}_c - \vec{t} \quad (A-5)$$

いま, スクリーン (画像) 上で直交する h 軸 (横方向) と v 軸 (縦方向) にそれぞれ平行な実際の空間での単位ベクトルを $\vec{a}$, $\vec{b}$ とする. ただし,

$$\vec{a} \cdot \vec{b} = 0 \text{ かつ } |\vec{a}| = 1 \text{ かつ } |\vec{b}| = 1 \quad (A-6)$$

また, スクリーンが点 C' でカメラの軸線と直交していることから,

$$\vec{a} \cdot \vec{l}_c = 0 \text{ かつ } \vec{b} \cdot \vec{l}_c = 0 \quad (A-7)$$

(A-4) 式および $\vec{a}$, $\vec{b}$ を用いると, 画像上の画素数単位の座標 (h, v) で示された点 Q' の実際の空間に対する位置ベクトルをターゲットの中心 O を原点とする座標系で,

$$\vec{t} + (\alpha/L) \vec{l}_c + h' \beta_a \vec{a} + v' \beta_b \vec{b} \quad (A-8)$$

と表すことができる. ここで,

$$h' = h - H/2, \quad v' = v - V/2 \quad (A-9)$$

とする. また, β_a と β_b はそれぞれ h 軸と v 軸方向の画素の辺の長さ (ピクセルサイズ) で, 与えられたデータから算出する (後述).

A.5 ターゲット上の点とスクリーン上の点の対応関係

スクリーン上の点 Q' に対応するターゲット上の点 Q の経度と緯度を λ , ϕ とすると, ターゲットの中心に原点をおく座標系で Q の位置ベクトルは,

$$\vec{r} = R \begin{pmatrix} \cos \phi & \cos \lambda \\ \cos \phi & \sin \lambda \\ \sin \phi \end{pmatrix} \quad (A-10)$$

と表すことができる. カメラ S から Q を見通したスクリーン上の点が Q' であるから, γ を Q (もしくは Q') ごとに異なる比例定数として, (A-8) 式より

$$\begin{aligned} \vec{l} &= \vec{r} - \vec{t} \\ &= \gamma \{ (\alpha/L) \vec{l}_c + h' \beta_a \vec{a} + v' \beta_b \vec{b} \} \\ &= \gamma' (\vec{l}_c + h' \beta'_a \vec{a} + v' \beta'_b \vec{b}) \end{aligned} \quad (A-11)$$

という関係があるはずである. ただし,

$$\gamma' = \gamma \alpha / L, \quad \beta'_a = \beta_a L / \alpha, \quad \beta'_b = \beta_b L / \alpha \quad (A-12)$$

である.

A.6 未知パラメータの算出

対応するターゲットの表面の点 Q_k がわかっているスクリーンの四隅の点 Q'_k のデータ ($k = 1, 2, 3, 4$) を用いて未知パラメータ β'_a , β'_b , $\vec{a}$, $\vec{b}$ を算出する. ただし, 与えられた点 Q_k のデータより,

$$\vec{r}_k = R \begin{pmatrix} \cos \phi_k & \cos \lambda_k \\ \cos \phi_k & \sin \lambda_k \\ \sin \phi_k \end{pmatrix} \quad (A-13)$$

$$\vec{l}_k = \vec{r}_k - \vec{t}$$

とする.

まず (A-11) 式の両辺と $\vec{a}$, $\vec{b}$ との内積をそれぞれ計算して, (A-6) 式および (A-7) 式より,

$$\vec{l} \cdot \vec{a} = \gamma' h' \beta'_a, \quad \vec{l} \cdot \vec{b} = \gamma' v' \beta'_b \quad (A-14)$$

が得られる. これらより γ' を消去して,

$$(v' \beta'_b \vec{a} - h' \beta'_a \vec{b}) \cdot \vec{l} = 0 \quad (A-15)$$

さらに (A-13) 式から,

$$\begin{aligned} (V \beta'_b \vec{a} - H \beta'_a \vec{b}) \cdot \vec{l}_1 &= 0, \\ (V \beta'_b \vec{a} + H \beta'_a \vec{b}) \cdot \vec{l}_2 &= 0, \\ (V \beta'_b \vec{a} - H \beta'_a \vec{b}) \cdot \vec{l}_3 &= 0, \\ (V \beta'_b \vec{a} + H \beta'_a \vec{b}) \cdot \vec{l}_4 &= 0 \end{aligned} \quad (A-16)$$

となる. ここで,

$$\vec{l}_{13} = \vec{l}_1 \times \vec{l}_3, \quad \vec{l}_{24} = \vec{l}_2 \times \vec{l}_4 \quad (A-17)$$

とすると（ただし、 $\times$ はベクトルの外積の記号）、(A-16) 式をまとめて、

$$\begin{aligned} V\beta'_b \vec{a} - H\beta'_a \vec{b} &= \chi \vec{l}_{13}, \\ V\beta'_b \vec{a} + H\beta'_a \vec{b} &= \xi \vec{l}_{24} \end{aligned} \quad (\text{A-18})$$

と書ける。ただし、 χ と ξ は未知の比例定数である。この式より、

$$\begin{aligned} \vec{a} &= \{1/(2V\beta'_b)\} (\xi \vec{l}_{24} + \chi \vec{l}_{13}), \\ \vec{b} &= \{1/(2H\beta'_a)\} (\xi \vec{l}_{24} - \chi \vec{l}_{13}) \end{aligned} \quad (\text{A-19})$$

となる。そして、 $\vec{a}$ と $\vec{b}$ の直交関係 (A-6) 式より、

$$\begin{aligned} \vec{a} \cdot \vec{b} &= \{1/(4HV\beta'_a\beta'_b)\} (\xi^2 |\vec{l}_{24}|^2 - \chi^2 |\vec{l}_{13}|^2) \\ &= 0 \end{aligned} \quad (\text{A-20})$$

であるから、 ξ を適当な定数として、

$$\chi = \xi/|\vec{l}_{13}|, \quad \xi = \xi/|\vec{l}_{24}| \quad (\text{A-21})$$

と書くことができる。この結果、(A-19) 式は、

$$\begin{aligned} \vec{a} &= \{\xi/(2V\beta'_b)\} (\vec{l}_{24}/|\vec{l}_{24}| + \vec{l}_{13}/|\vec{l}_{13}|), \\ \vec{b} &= \{\xi/(2H\beta'_a)\} (\vec{l}_{24}/|\vec{l}_{24}| - \vec{l}_{13}/|\vec{l}_{13}|) \end{aligned} \quad (\text{A-22})$$

となる。さらに、 $|\vec{a}| = 1$ かつ $|\vec{b}| = 1$ ((A-6) 式) から、 $\vec{a}$ 、 $\vec{b}$ を

$$\begin{aligned} \vec{a} &= (\vec{l}_{24}/|\vec{l}_{24}| + \vec{l}_{13}/|\vec{l}_{13}|)/ \\ &\quad |\vec{l}_{24}/|\vec{l}_{24}| + \vec{l}_{13}/|\vec{l}_{13}||, \\ \vec{b} &= (\vec{l}_{24}/|\vec{l}_{24}| - \vec{l}_{13}/|\vec{l}_{13}|)/ \\ &\quad |\vec{l}_{24}/|\vec{l}_{24}| - \vec{l}_{13}/|\vec{l}_{13}|| \end{aligned} \quad (\text{A-23})$$

と表すことができる。

以上で与えられたデータからパラメータ $\vec{a}$ 、 $\vec{b}$ が計算できた。次に β'_a 、 β'_b を求める。そのために、(A-14) 式の場合と同様に、(A-11) 式の両辺と $\vec{l}_c$ の内積をとると、 $|\vec{l}_c| = L$ だから、

$$\vec{l} \cdot \vec{l}_c = r' L^2 \quad (\text{A-24})$$

これと (A-14) 式とをあわせて r' を消去すると、

$$\begin{aligned} h'\beta'_a \vec{l} \cdot \vec{l}_c - L^2 \vec{l} \cdot \vec{a} &= 0, \\ v'\beta'_b \vec{l} \cdot \vec{l}_c - L^2 \vec{l} \cdot \vec{b} &= 0 \end{aligned} \quad (\text{A-25})$$

したがって、

$$\begin{aligned} \beta'_a &= (L^2/h') (\vec{a} \cdot \vec{l}) / (\vec{l}_c \cdot \vec{l}), \\ \beta'_b &= (L^2/v') (\vec{b} \cdot \vec{l}) / (\vec{l}_c \cdot \vec{l}) \end{aligned} \quad (\text{A-26})$$

これらに (A-13) 式で与えられる $\vec{l} = \vec{l}_k$ ($k = 1, 2, 3, 4$) のいずれか（そしてそれに対応する h' 、 v' の値）を代入すれば β'_a 、 β'_b が計算できる。ここでは対称性を考えて、それらの計算値の平均を用いることにする。

$$\begin{aligned} \beta'_a &= \{L^2/(2H)\} \{-(\vec{a} \cdot \vec{l}_1)/(\vec{l}_c \cdot \vec{l}_1) \\ &\quad + (\vec{a} \cdot \vec{l}_2)/(\vec{l}_c \cdot \vec{l}_2) \\ &\quad + (\vec{a} \cdot \vec{l}_3)/(\vec{l}_c \cdot \vec{l}_3) \\ &\quad - (\vec{a} \cdot \vec{l}_4)/(\vec{l}_c \cdot \vec{l}_4)\}, \\ \beta'_b &= \{L^2/(2V)\} \{-(\vec{b} \cdot \vec{l}_1)/(\vec{l}_c \cdot \vec{l}_1) \\ &\quad - (\vec{b} \cdot \vec{l}_2)/(\vec{l}_c \cdot \vec{l}_2) \\ &\quad + (\vec{b} \cdot \vec{l}_3)/(\vec{l}_c \cdot \vec{l}_3) \\ &\quad + (\vec{b} \cdot \vec{l}_4)/(\vec{l}_c \cdot \vec{l}_4)\} \end{aligned} \quad (\text{A-27})$$

A.7 スクリーン上の任意の点に対応するターゲット上の点の位置の計算

画素数単位の座標値 (h' 、 v') で表されたスクリーン上の任意の点 Q' に対応するターゲット上の点 Q の位置ベクトル $\vec{r}$ は、(A-11) 式を変形して得た

$$\vec{q} = \vec{l}_c + h'\beta'_a \vec{a} + v'\beta'_b \vec{b}, \quad \vec{r} = \vec{t} + r' \vec{q} \quad (\text{A-28})$$

の比例定数 r' を求めれば計算することができる。(A-28) 式の 2 番目の式の両辺の成分の自乗和をとると、 $|\vec{r}| = R$ 、 $|\vec{t}| = T$ より、

$$|\vec{q}|^2 r'^2 + 2(\vec{q} \cdot \vec{t}) r' + T^2 - R^2 = 0 \quad (\text{A-29})$$

これを解けば r' に関する 2 つの根が得られる。このうち、値の小さな方が求める比例定数である（中野，1995）。

$$\begin{aligned} I &= \vec{q} \cdot \vec{t}, \\ r' &= -[I + \{I^2 - |\vec{q}|^2(T^2 - R^2)\}^{1/2}]/|\vec{q}|^2 \end{aligned} \quad (\text{A-30})$$

A.8 ターゲット上の任意の点に対応するスクリーン上の点の位置の計算

位置ベクトル $\vec{r}$ で表されたターゲット上の点 Q のスクリーン上の投影点の位置 (h' と v') は、(A-25) 式から得られる式によって計算することができる。

$$\begin{aligned} h' &= (L^2/\beta'_a) (\vec{a} \cdot \vec{l}) / (\vec{l}_c \cdot \vec{l}), \\ v' &= (L^2/\beta'_b) (\vec{b} \cdot \vec{l}) / (\vec{l}_c \cdot \vec{l}) \end{aligned} \quad (\text{A-31})$$

A.9 画像の位置決めの手順（まとめ）

- (1) 与えられたデータを (A-1)、(A-2)、(A-3) および (A-5) 式に代入して、パラメータ $\vec{l}$ 、 R および $\vec{l}_c$ を計算する。
- (2) 与えられたデータと上で得たパラメータを (A-13)、(A-17)、(A-23) 式に代入して、パラメータ $\vec{l}_k$ ($k = 1, 2, 3, 4$)、 $\vec{a}$ 、 $\vec{b}$ を計算する。
- (3) 与えられたデータと上で得たパラメータを (A-27) 式に代入して、パラメータ β'_a および β'_b を求めておく。
- (4) スクリーン上の点の画素単位の座標値 (h 、 v) からそれに対応するターゲット上の点の経度 λ と緯度 ϕ を計算したい場合には、まず与えられたデータおよび上で求めたパラメータを用いて (A-9)、(A-28)、(A-30) 式から r' を計算する。その結果を (A-28) 式に代入して得た $\vec{r}$ は (A-10) 式のように表されるので、それを解いて λ と ϕ を求める。
- (5) ターゲット上の経度 λ 、緯度 ϕ の点に対応する画像上の点の画素単位の座標値 (h 、 v) を計算する場合には、与えられたデータおよび先に求めたパラメータを用いて (A-31) および (A-9) 式を解くことになる。

付録B. Clementine 画像のステレオ視

以下では、CD-ROM として配布された Clementine 画

像のペアをステレオ視するために必要なパラメータの導出法について説明する。ただし、取りあつかう画像はすべて、付録 A で説明したようにして、画像（スクリーン）上の点の座標とそこに対応するターゲット（月）上の点の座標の対応関係がわかっている（すなわち、画像の位置決めができてい）ものとする。なお、ステレオ視を行いたい画像（ペア）を撮影した時の Clementine 探査機（カメラ）とターゲット上の点の位置関係や以下の説明で用いている記号などについては第 5 図を参照していただきたい。

B.1 仮定

- (1) ステレオ視の可能性のチェックおよびステレオ視のためのパラメータの計算を行う Clementine 画像のペアの一方を Master 画像、他方を Slave 画像と呼ぶ。ただし、どちらの画像を Master 画像（もしくは Slave 画像）にするかは任意である。
- (2) ステレオ視のために作成するものは、オリジナルの Master 画像とオリジナルを単に拡大・回転した Slave 画像を並べた画像であるとする（画像を取得した時のカメラの位置や太陽高度などの違いに起因する Slave 画像の歪みや輝度値などの補正はここでは考慮しない）。ただし、両者の並べ方、すなわち Master 画像に対する Slave 画像の移動方向は計算する。
- (3) Master 画像と Slave 画像のどちらにも含まれているターゲット上の領域（共通領域）に相当する Master 画像上の領域の重心を参照点と呼ぶ。ステレオ視を行うために必要な Master 画像に対する Slave 画像の拡大率・回転角度・移動方向の計算、そして、ステレオ視が可能か否かを判定する際に重要なそれぞれの画像に対するカメラからターゲット上の点へ向かうベクトル（視線ベクトル）の方向の違い（視線ベクトルのなす角）の計算は、参照点もしくはそれに対応するターゲット上の点について行うものとする。

B.2 画像の位置決めに必要なパラメータ

（付録 A を参照のこと）

$\vec{t}$:

ターゲットの中心を原点としたカメラ P の位置ベクトル。
 $\vec{a}$, $\vec{b}$:

ともにカメラの軸線に直交し、それぞれ画像の横・縦方向に平行な単位ベクトル。ただし、カメラ P からカメラの軸線（画像（スクリーン）の中央の点を通ると仮定）がターゲットの表面と交わる点へと向かうベクトルを $\vec{l}_c$ として、

$$\begin{aligned} \vec{a} \cdot \vec{l}_c = 0 \text{ かつ } \vec{b} \cdot \vec{l}_c = 0 \text{ かつ } \vec{a} \cdot \vec{b} = 0 \text{ かつ } \\ |\vec{a}| = 1 \text{ かつ } |\vec{b}| = 1 \end{aligned} \quad (\text{B-1})$$

が成り立っているものとする。

γ' , β'_a , β'_b :

画像（スクリーン）上の点 Q' の座標とそこに対応するターゲット上の点 Q の座標の間の関係式に現われる係数、画像の横・縦画素数を H と V、画像上の点 Q' の位置を画素単位で (h, v)、また、そこに対応するターゲット上の点 Q のターゲットの中心を原点とする位置ベクトルを $\vec{r}$ とすると、この関係式は、

$$\begin{aligned} h' &= h - H/2, \quad v' = v - V/2, \\ \vec{l} &= \vec{r} - \vec{t} = \gamma' (\vec{l}_c + \beta'_a h' \vec{a} + \beta'_b v' \vec{b}) \end{aligned} \quad (\text{B-2})$$

となる。ここで、 $\vec{l} = \vec{r} - \vec{t}$ は、カメラ P からターゲット上の点 Q に向かう視線ベクトルである。なお、 β'_a , β'_b はスクリーンのすべての点で同じ値をとる係数であるが、 γ' は点ごとに異なった値となる。

なお、以下では、これらのパラメータなどに添え字 M と S をつけて Master 画像と Slave 画像に関するものであることを区別する。

B.3 画像の共通領域のチェック

第 5 (b) 図に示されているような Master 画像と Slave 画像のどちらにも含まれているターゲット上の領域（共通領域）を調べて、以下のように定義される画像の重なり度合い δ を計算する。

$$\delta = \frac{\text{Master 画像上での共通領域の面積 (画素数)}}{\text{Master 画像の面積 (画素数)}} \quad (\text{B-3})$$

共通領域の面積のカウントは、Master 画像のすべての画素をスキャンしてそれに対応するターゲット上の点の座標を (B-2) 式を使って計算し、その点に対応する Slave 画像上の点の有無を ((B-2) 式を用いて) 調べることで行う。なお、 δ の値が 0 の画像ペアはステレオ視が不可能である（ターゲット上の同じ領域を撮影していない）ので、これ以降の処理は行わない。

上の処理と平行して、Master 画像上での共通領域の重心（参照点） Q'_M の位置を計算する。また、そこに対応するターゲット上の点 Q の座標と、さらに、それから計算できる Slave 画像上の参照点 Q'_S の位置を求めておく。

なお、これ以降の処理では、(B-2) 式を利用した計算は参照点に関するものだけを行うので、そこに現われる係数 γ' （画像上の点ごとに異なる係数）の値として参照点に関する値だけを用いることになる。

B.4 視線ベクトル間の角度

Master 画像および Slave 画像のカメラ P_M および P_S から参照点に対応するターゲット上の点 Q に向かう視線ベクトルをそれぞれ $\vec{l}_M$, $\vec{l}_S$ とすると（第 5 (a) 図）、これら

のなす角度 θ は,

$$\theta = \cos^{-1} \{ (\vec{l}_M \cdot \vec{l}_S) / (|\vec{l}_M| |\vec{l}_S|) \} \quad (B-4)$$

となる. なお, θ の値が 0 の画像ペアはステレオ視が不可能なので, これ以降の処理は行わない.

B.5 Master 画像に対する Slave 画像の移動方向

ステレオ視では, 実際には異なった位置にあるカメラ P_M と P_S から見たターゲット上の点 Q (第 5(a)図) を, 同じ位置にあるカメラ P から見たターゲット上の異なった点 Q_M と Q_S であるとするように Master 画像と Slave 画像を配置する (第 5(c)図). その際, 点 Q_M に対する Q_S の移動量 $\vec{d}$ は, カメラ P_M に対する P_S の位置ベクトルの相違 $\vec{t}_S - \vec{t}_M$ に等しい.

$$\vec{d} = \vec{t}_S - \vec{t}_M \quad (B-5)$$

ステレオ視のために Master 画像と Slave 画像を並べるとき (第 5(d)図), Master 画像上の参照点に対する Slave 画像上の参照点の移動方向としてベクトル $\vec{d}$ の Master 画像のスクリーンに平行な成分 $\vec{d}'$ の方向をとればよい. $\vec{d}'$ は, Master 画像のスクリーン上の単位ベクトル $\vec{a}_M$ と $\vec{b}_M$ を用いて,

$$\vec{d}' = (\vec{d} \cdot \vec{a}_M) \vec{a}_M + (\vec{d} \cdot \vec{b}_M) \vec{b}_M \quad (B-6)$$

と表すことができる. さらに, ステレオ視の際の Master 画像に対する Slave 画像の移動方向を Master 画像の右方向を 0, 下方向を 90 度とする方位角 ω を用いて表すと,

$$\omega = \tan^{-1} \{ (\vec{d} \cdot \vec{b}_M) / (\vec{d} \cdot \vec{a}_M) \} \quad (B-7)$$

となる.

なお, ステレオ視の際の Master 画像上の参照点に対する Slave 画像上の参照点の移動距離も $|\vec{d}'|$ に相当する値にすべきなのかもしれないが, 人間の目にはかなりのフレキシビリティがあるので実際にはそこまで厳密に考えなくてもよいようである.

B.6 Master 画像に対する Slave 画像の拡大率

通常は, Master 画像と Slave 画像とで画像の解像度 (画素の一辺に相当するターゲット上での実際の長さ) が

異なっているため, これらを並べてステレオ視するためには Slave 画像を拡大 (もしくは縮小) しなければならない. (B-2) 式から画像の解像度 Δ は, 横方向のものと縦方向のものの平均をとって,

$$\Delta = (\gamma'/2) (\beta'_a + \beta'_b) \quad (B-8)$$

と表すことができる. これより, Master 画像に対する Slave 画像の拡大率 m は,

$$m = \Delta_S / \Delta_M \quad (B-9)$$

となる.

なお, オリジナルの画像の精度を活かすためには, 画像の縮小は避けるべきである. すなわち, m の値が 1 より小さい画像ペアの場合には, Slave 画像を縮小するのではなく Master 画像を $1/m$ 倍に拡大するほうがよい.

B.7 Master 画像に対する Slave 画像の回転角度

通常はターゲット上の方向 (たとえば月の子午線の北向きの方向) が Master 画像と Slave 画像で異なった方向に投影されているので, ステレオ視を行うためには Slave 画像を回転させたものを Master 画像の横に配置してやる必要がある (第 5(b)図). Master 画像に対する Slave 画像の回転は, Slave 画像のスクリーン上の単位ベクトル $\vec{a}_S$ (もしくは $\vec{b}_S$) の Master 画像のスクリーン上への投影 $\vec{a}'_S$ (もしくは $\vec{b}'_S$) と Master 画像の単位ベクトル $\vec{a}_M$ (もしくは $\vec{b}_M$) のなす角度だけ行う. すなわち,

$$\begin{aligned} \vec{a}'_S &= (\vec{a}_S \cdot \vec{a}_M) \vec{a}_M + (\vec{a}_S \cdot \vec{b}_M) \vec{b}_M, \\ \vec{b}'_S &= (\vec{b}_S \cdot \vec{a}_M) \vec{a}_M + (\vec{b}_S \cdot \vec{b}_M) \vec{b}_M \end{aligned} \quad (B-10)$$

なので, $\vec{a}_M$ と $\vec{a}'_S$ のなす角度 ε_a および $\vec{b}_M$ と $\vec{b}'_S$ のなす角度 ε_b はそれぞれ, Master 画像の右方向を 0, 下方向を 90 度とする回転方向で,

$$\begin{aligned} \varepsilon_a &= \tan^{-1} \{ (\vec{a}_S \cdot \vec{b}_M) / (\vec{a}_S \cdot \vec{a}_M) \}, \\ \varepsilon_b &= \tan^{-1} \{ (\vec{b}_S \cdot \vec{b}_M) / (\vec{b}_S \cdot \vec{a}_M) \} \end{aligned} \quad (B-11)$$

となる. ここでは, これらの平均をとって, Master 画像に対する Slave 画像の回転角度 ε を

$$\varepsilon = (1/2) (\varepsilon_a + \varepsilon_b) \quad (B-12)$$

とする.

(1995年7月10日受付, 1995年9月20日受理)

付録C. プログラムのソースリスト

```

link.cdrom (C-Shell script) : execute after "chmod +x link.cdrom"
1: #!/bin/csh
2: # usage : link.cdrom srcDir dstDir
3: #
4: #alias stderr echo # comment out not on my system
5: #
6: if ($#argv != 2) then
7:   stderr usage : link.cdrom srcDir dstDir ; exit 1
8: endif
9: #
10: set src=$argv[1]
11: set dst=$argv[2]
12: #
13: foreach org ('ls $src')
14:   set new=`echo $org | tr 'A-Z' 'a-z'` | awk -F' ' '{ print $1 }'`
15: #
16:   if (-d $src)/$org then
17:     mkdir ${dst}/${new} ; link.cdrom ${src}/${org} ${dst}/${new}
18:   else
19:     ln -s ${src}/${org} ${dst}/${new}
20:   endif
21: end

linkcdrom.c : compile on UNIX as "cc linkcdrom.c -o linkcdrom"
1: #include <stdio.h>
2: #include <sys/types.h>
3: #include <sys/dir.h>
4: #include <sys/stat.h>
5: #include <sys/stat.h>
6:
7: static void Error(msg)
8:   char *msg;
9: {
10:   fputs(msg,stderr); fputc('\n',stderr); exit(1);
11: }
12:
13: #define NAME structDirect->d_name
14:
15: static void LinkCDROM(src,dst)
16:   char *src,*dst;
17: {
18:   int srcLen=strlen(src),dstLen=strlen(dst),len,idx;
19:   DIR *dir;
20:   struct direct *structDirect;
21:   struct stat structStat;
22:   char *org,*new;

23:   if ((dir=opendir(src))==NULL) Error("directory read error.");
24:
25:   while ((structDirect=readdir(dir))!=NULL)
26:     if (strcmp(NAME,".") && strcmp(NAME,"..")) {
27:       len=strlen(NAME);
28:       if ((org=(char *)malloc(srcLen+len+2))==NULL ||
29:           (new=(char *)malloc(dstLen+len+2))==NULL)
30:         Error("name area allocation error.");
31:       sprintf(org,"%s/%s",src,NAME);
32:       if (stat(org,&structStat)==(-1)) Error("status read error.");
33:
34:       sprintf(new,"%s/%s",dst,NAME);
35:       for (idx=1; idx<=len; idx++)
36:         if (new[idx+dstLen]!='\0') {
37:           new[idx+dstLen]='\0'; break;
38:         }
39:       else
40:         new[idx+dstLen]='\0';
41:
42:       if ((structStat.st_mode&S_IFMT)==S_IFDIR) {
43:         if (mkdir(new,S_IRUSR|S_IWUSR|S_IXUSR|S_IRGRP|S_IWGRP|S_IXGRP|S_IROTH|S_IWOTH)==(-1))
44:           Error("make directory error.");
45:
46:         LinkCDROM(org,new);
47:       }
48:       else
49:         if (symlink(org,new)==(-1))
50:           fprintf(stderr,"symbolic link error for %s.\n",org);
51:         free(org); free(new);
52:       }
53:     }
54:   closedir(dir);
55:
56:   main(argc,argv)
57:     argc;
58:     **argv;
59:   {
60:     if (argc!=3) Error("usage : linkcdrom cdromDirectory directory");
61:     LinkCDROM(argv[1],argv[2]);
62:   }

```



```

1: #include <stdio.h>
2: static void Error(msg)
3:     *msg;
4: {
5:     fputs(msg,stderr); fputc('\n',stderr); exit(1);
6: }
7:
8: #define STRLEN 256
9: static int Skip(file)
10:     *file;
11: {
12:     int chr;
13:     while ((chr=fgetc(file))!=' ' || chr=='\t' || chr=='\n');
14:     if (chr=='#') {
15:         while ((chr=fgetc(file))!=EOF && chr!='\n');
16:         return(Skip(file));
17:     }
18:     return(chr);
19: }
20: static int GetNum(file)
21:     *file;
22: {
23:     int chr=Skip(file),num=0;
24:     if (chr==EOF) return(-1);
25:     if (chr<'0' || chr>'9') Error("not a numeric character.");
26:     do num=num*10+chr-'0'; while ((chr=fgetc(file))>='0' && chr<='9');
27:     ungetc(chr,file); return(num);
28: }
29: static void PutStr(str,bgn,len)
30:     *str;
31:     bgn,len;
32: {
33:     str=(str[bgn]);
34:     while (--len>0 && str[len]!=' ');
35:     for (bgn=0; bgn<len && str[bgn]!=' '; bgn++);
36:     while (bgn<len) putchar(str[bgn++]);
37: }
38: #define List struct LIST
39: List {
40:     int bgn,len;
41:     List *next;
42: };
43: #define DELIM '\t'
44: int main(argc,argv)
45: {
46:     FILE *file;
47:     int size,bgn,len;
48:     char *rec;
49:     List *list,*ptr,*top=NULL;
50:     if (argc!=3) Error("usage : tab2tbl script TAB");
51:     if (argv[1][0]!='-')
52:         file=stdin;
53:     else
54:         if ((file=fopen(argv[1],"r"))==NULL)
55:             Error("script file not found.");
56:     if ((size=GetNum(file))<=0) Error("bad record size.");
57:     if ((rec=(char *)malloc(size))==NULL)
58:         Error("memory for record buffer allocation error.");
59:     while ((bgn=GetNum(file))>=0 && (len=GetNum(file))>=0) {
60:         if (bgn==0 && len==0) break;
61:         if (bgn==0 || len==0 || bgn+len-1>size)
62:             Error("bad record item location.");
63:         if ((list=(List *)malloc(sizeof(*list)))==NULL)
64:             Error("memory for record item list allocation error.");
65:         list->bgn=bgn-1; list->len=len; list->next=NULL;
66:         if (top==NULL) top=ptr=list; else ptr=(ptr->next=list);
67:         if (file!=stdin) fclose(file);
68:         if (top==NULL) Error("no record item specified.");
69:         if (argv[2][0]!='-')
70:             file=stdin;
71:         else
72:             if ((file=fopen(argv[2],"rb"))==NULL)
73:                 Error("TAB file not found.");
74:         while (fread(rec,size,1,file)==1) {
75:             for (list=top; list!=NULL; list=list->next) {
76:                 if (list->top) putchar(DELIM);
77:                 PutStr(rec,list->bgn,list->len);
78:                 putchar('\n');
79:             }
80:             if (file!=stdin) fclose(file);
81:         }
82:     }
83: }

```

```

count.cf (C-Shell script) : execute after "chmod +x count.cf"
1: #!/bin/csh
2: # usage : count.cf cameraName filterName
3: #
4: #alias stderr echo # comment out not on my system
5: #alias awk gawk # comment out if GNU awk is available

```

```

6: #
7: if ($argv != 2) then
8:   stderr usage : count.cf cameraName filterName ; exit 1
9: endif
10: #
11: set tab=imindx.tab # change this as suitable for your environment
12: if (!(-e $tab)) then
13:   stderr .TAB file $tab not found. ; exit 1
14: endif
15: #
16: set camera=$argv[1]
17: set filter=$argv[2]
18: #
19: cat <<'fin'>count.cf.awk
20: BEGIN {
21:   camera = "$camera"
22:   filter = "$filter"
23: }
24: #
25: cat <<'fin'>count.cf.awk
26: total = 0 # total image number
27: moon = 0 # number of TARGET=MOON images
28: }
29: $1 == camera && $2 == filter {
30:   ++total
31:   if ($3 == "MOON") ++moon
32: }
33: END {
34:   printf "%s\t%s\t%d\t%d\n", camera, filter, total, moon
35: }
36: 'fin'
37: #
38: cat <<'fin' | tab2tbl - $tab | awk -f count.cf.awk ; rm count.cf.awk
39: 882 # record size of "imindx.tab"
40: 28 6 # 1. INSTRUMENT_ID
41: 128 1 # 2. FILTER_NAME
42: 236 5 # 3. TARGET_NAME
43: 'fin'

search.loc (C-Shell script) : execute after "chmod +x search.loc"

1: #!/bin/csh
2: # usage : search.loc longi. lati.
3: #
4: #alias stderr echo # comment out not on my system
5: #alias awk gawk # comment out if GNU awk is available
6: #
7: if ($argv != 2) then
8:   stderr usage : search.loc longi. lati. ; exit 1
9: endif
10: #
11: set tab=imindx.tab # change this as suitable for your environment
12: if (!(-e $tab)) then
13:   stderr .TAB file $tab not found. ; exit 1
14: endif
15: #
16: set lon=$argv[1]
17: set lat=$argv[2]
18: #
19: cat <<'fin'>search.loc.awk
20: function toX(lon, lat) { return cos(lat) * cos(lon) }
21: function toY(lon, lat) { return cos(lat) * sin(lon) }

22: function toZ( lat) { return sin(lat) }
23: BEGIN {
24:   rad = 0.01745329251994329576 # = PI / 180
25: 'fin'
26: #
27: cat <<'fin'>search.loc.awk
28: x0 = toX($lon * rad, $lat * rad)
29: y0 = toY($lon * rad, $lat * rad)
30: z0 = toZ(
31: 'fin'
32: #
33: cat <<'fin'>search.loc.awk
34: }
35: function side(x0, y0, z0, x1, y1, z1, x2, y2, z2) {
36:   return x0 * (y1 * z2 - z1 * y2) + \
37:     y0 * (z1 * x2 - x1 * z2) + \
38:     z0 * (x1 * y2 - y1 * x2)
39: }
40: $1 == "MOON" {
41:   x1 = toX($2 * rad, $3 * rad)
42:   y1 = toY($2 * rad, $3 * rad)
43:   z1 = toZ(
44:   x2 = toX($4 * rad, $5 * rad)
45:   y2 = toY($4 * rad, $5 * rad)
46:   z2 = toZ(
47:   if (side(x0, y0, z0, x1, y1, z1, x2, y2, z2) <= 0) {
48:     x3 = toX($6 * rad, $7 * rad)
49:     y3 = toY($6 * rad, $7 * rad)
50:     z3 = toZ(
51:   if (side(x0, y0, z0, x2, y2, z2, x3, y3, z3) <= 0) {
52:     x4 = toX($8 * rad, $9 * rad)
53:     y4 = toY($8 * rad, $9 * rad)
54:     z4 = toZ(
55:   if (side(x0, y0, z0, x3, y3, z3, x4, y4, z4) <= 0) {
56:     if (side(x0, y0, z0, x4, y4, z4, x1, y1, z1) <= 0) {
57:       printf "%s", $10, $11
58:       for (idx = 12 ; idx <= NF ; idx++) printf "\t%s", $idx
59:       printf "\n"
60:     }
61:   }
62: }
63: }
64: 'fin'
65: #
66: cat <<'fin' | tab2tbl - $tab | awk -f search.loc.awk ; rm search.loc.awk
67: 882 # record size of "imindx.tab"
68: 236 5 # 1. TARGET_NAME
69: 445 7 # 2. RETICLE_POINT_LONGITUDE_1
70: 437 7 # 3. RETICLE_POINT_LATITUDE_1
71: 461 7 # 4. RETICLE_POINT_LONGITUDE_2
72: 453 7 # 5. RETICLE_POINT_LATITUDE_2
73: 477 7 # 6. RETICLE_POINT_LONGITUDE_3
74: 469 7 # 7. RETICLE_POINT_LATITUDE_3
75: 493 7 # 8. RETICLE_POINT_LONGITUDE_4
76: 485 7 # 9. RETICLE_POINT_LATITUDE_4
77: 781 25 # 10. PATH_NAME
78: 183 12 # 11. FILE_NAME
79: 11 3 # 12. REVOLUTION_NUMBER
80: 28 6 # 13. INSTRUMENT_ID
81: 128 1 # 14. FILTER_NAME
82: 'fin'

```